AMENDMENTS TO THE CLAIMS

1-65. (Canceled)

66. (Currently amended) A computerized method for ~~solving a constraint satisfaction problem~~ controlling operation of a target system, comprising:

receiving a set of variables that are characteristic of inputs to the target system, the variables having respective input domains, and a set of constraints comprising one or more relations defined as a combination of operators and expressed in a generic relational language, the combination comprising at least one operator selected from a group of arithmetic and bitwise operators, which are applied to the variables;

automatically parsing the constraints expressed in the generic relational language so as to build~~ing~~ a network ~~of~~ comprising one or more hyper-arcs ~~representing the set of constraints~~, each hyper-arc corresponding to ~~one of the relations expressed in terms of the operators and~~ a respective relation and comprising a constraint sub-network linking nodes in the network corresponding to the variables to which the operators are applied; ~~and~~

processing each constraint sub-network so as to find sets of values of the variables within the input domains that satisfy the respective relation;

reducing the input domains of the variables in the network ~~responsive to the operators~~ using the sets of values of the variables, so as to determine respective output domains of the variables that are consistent with the set of constraints of the network; and

determining values of the inputs to be made to the target system based on the consistent output domains of the variables.

67. (Canceled)

68. (Previously presented) A method according to claim 66, wherein reducing the input domains comprises finding projections of the operators onto the domains of the variables.

69. (Currently amended) A method according to claim ~~66~~ 68, wherein receiving the set of variables comprises receiving an output variable for each of the operators variable and at least one input variable for each of the operators, and

wherein finding the projections comprises projecting the domain of the at least one input variable of each of the operators onto the domain of the output variable thereof, and projecting the domain of the output variable of each of the operators onto the domain of the at least one input variable thereof.

70. (Previously presented) A method according to claim 66, wherein the operators comprise multi-variable operators, which receive two of more of the variables as their inputs.

71. (Previously presented) A method according to claim 70, wherein the multi-variable operators comprise one or more operators selected from a group consisting of arithmetic addition, arithmetic subtraction, arithmetic multiplication, arithmetic division and modulo operators.

72. (Previously presented) A method according to claim 70, wherein the multi-variable operators comprise one or more operators selected from a group consisting of an operator testing

arithmetic equality of two of the variables, an operator testing arithmetic inequality of two of the variables, and an operator testing whether one of the variables is greater than another of the variables.

73. (Previously presented) A method according to claim 70, wherein the multi-variable operators comprise one or more operators selected from a group consisting of a bitwise "and," bitwise "or" and bitwise "exclusive or" operations.

74. (Previously presented) A method according to claim 66, wherein the combination of operators further comprises at least one logical operator.

75. (Previously presented) A method according to claim 74, wherein the at least one logical operator comprises at least one of a disjunction operator, a negation operator, and an implication operator.

76. (Currently amended) A method according to claim 66, wherein ~~reducing the input domains~~ processing each constraint sub-network comprises, for each of the hyper-arcs, assembling the variables in a hierarchy based on the relation corresponding to the hyper-arc, and reducing the input domains of the variables in the hierarchy.

77. (Previously presented) A method according to claim 76, wherein assembling the variables comprises arranging the variables in a hierarchical graph, having vertices corresponding to the variables.

78. (Previously presented) A method according to claim 77, wherein arranging the variables in the hierarchical graph comprises arranging the graph so as to have the form of one or more trees.

79. (Currently amended) A method according to claim 78, wherein ~~reducing the input domains~~ processing each constraint sub-network comprises reducing the input domains over each of the trees so as to find respective interim domains of the variables that are consistent with the relation over each of the trees, and combining the interim domains over all of the trees to determine the output domains.

80. (Previously presented) A method according to claim 77, wherein arranging the variables in the graph comprises inserting vertices in the graph corresponding to the operators, connecting the vertices corresponding to the variables.

81. (Currently amended) A method according to claim 66, wherein ~~building the network of the hyper-arcs~~ automatically parsing the constraints comprises representing the set of relations as a disjunction of multiple relations, with one of the hyper-arcs corresponding respectively to each of the relations, and

wherein ~~determining the respective output~~ reducing the input domains comprises determining interim domains of the variables for each of the hyper-arcs, and taking a union of the interim domains for each of the variables to determine the output domains.

82. (Currently amended) A method according to claim 66, wherein ~~reducing the input domains~~ processing each constraint sub-network comprises ~~determining the output domains~~ finding the sets of values of the variables within the input domains such that for any given value in the respective

output domain of each of the variables, there exist values of the other variables in the respective output domains thereof that, together with the given value, constitute a solution to the set of relations.

83. (Currently amended) A method according to claim 66, wherein ~~reducing the input domains~~ processing each constraint sub-network comprises ~~determining the output domains~~ finding the sets of values of the variables within the input domains such that every set of values of the variables in the input domains that constitutes a solution to the set of relations is contained in the output domains of the variables.

84. (Canceled)

85. (Currently amended) A method according to claim ~~84~~ 66, wherein the system comprises an electronic processor, and wherein determining the values of the inputs comprises determining commands and addresses to be input to the processor.

86. (Currently amended) A method according to claim 66, wherein ~~receiving the set of variables comprises receiving control parameters of~~ the system comprises a mechanical system, and wherein ~~reducing the input domains~~ determining the values of the inputs comprises generating a command to control the system ~~based on the output domains of the parameters~~.

87-90. (Canceled)

91. (Previously presented) A method according to claim 66, wherein at least one of the constraints comprises a relation among at least three of the variables.

92. (Currently amended) Apparatus for ~~solving a constraint satisfaction problem~~ controlling operation of a target system, comprising a constraint processor, arranged to receive a set of variables that are characteristic of inputs to the target system, the variables having respective input domains, and a set of constraints comprising one or more relations defined as a combination of operators and expressed in a generic relational language, the combination comprising at least one operator selected from a group of arithmetic and bitwise operators, which are applied to the variables,

wherein the constraint processor is arranged to automatically parse the constraints expressed in the generic relational language so as to build a network ~~of~~ comprising one or more hyper-arcs ~~representing the set of constraints~~, each hyper-arc corresponding to ~~one of the relations expressed in terms of the operators and~~ a respective relation and comprising a constraint sub-network linking nodes in the network corresponding to the variables to which the operators are applied, to process each constraint sub-network so as to find sets of values of the variables within the input domains that satisfy the respective relation, ~~and~~ to reduce the input domains of the variables in the network ~~responsive to the operators~~ using the sets of values of the variables, so as to determine respective output domains of the variables that are consistent with the set of constraints, and to determine values of the inputs to be made to the target system based on the consistent output domains of the variables.

93. (Canceled)

94. (Previously presented) Apparatus according to claim 92, wherein the processor is arranged to reduce the input domains by finding projections of the operators onto the domains of the variables.

95. (Previously presented) Apparatus according to claim 94, wherein the set of variables comprises an output variable and at least one input variable for each of the operators, and wherein the processor is arranged to project the domain of the at least one input variable of each of the operators onto the domain of the output variable thereof, and to project the domain of the output variable of each of the operators onto the domain of the at least one input variable thereof.

96. (Previously presented) Apparatus according to claim 92, wherein the operators comprise multi-variable operators, which receive two of more of the variables as their inputs.

97. (Previously presented) Apparatus according to claim 96, wherein the multi-variable operators comprise one or more operators selected from the group consisting of arithmetic addition, arithmetic subtraction, arithmetic multiplication, arithmetic division and modulo operators.

98. (Previously presented) Apparatus according to claim 96, wherein the multi-variable operators comprise one or more operators selected from the group consisting of an operator testing arithmetic equality of two of the variables, an operator testing arithmetic inequality of two of the variables, and an operator testing whether one of the variables is greater than another of the variables.

99. (Previously presented) Apparatus according to claim 96, wherein the multi-variable operators comprise one or more operators selected from a group consisting of a bitwise "and," bitwise "or" and bitwise "exclusive or" operations.

100. (Previously presented) Apparatus according to claim 92, wherein the combination of operators further comprises at least one logical operator.

101. (Previously presented) Apparatus according to claim 100, wherein the at least one logical operator comprises at least one of a disjunction operator, a negation operator, and an implication operator.

102. (Previously presented) Apparatus according to claim 92, wherein the processor is arranged, for each of the hyper-arcs, to assemble the variables in a hierarchy based on the relation corresponding to the hyper-arc, and to reduce the input domains of the variables in the hierarchy, so as to determine the respective output domains of the variables that are consistent with the set of constraints.

103. (Previously presented) Apparatus according to claim 102, wherein the hierarchy of the variables comprises a hierarchical graph, having vertices corresponding to the variables.

104. (Previously presented) Apparatus according to claim 103, wherein the hierarchical graph has the form of one or more trees.

105. (Previously presented) Apparatus according to claim 104, wherein the processor is arranged to reduce the input domains over each of the trees so as to find respective interim domains of the

variables that are consistent with the relation over each of the trees, and to combine the interim domains over all of the trees to determine the output domains.

106. (Currently amended) Apparatus according to claim 103, ~~wherein the set of constraints is defined as a combination of operators, selected from a group of arithmetic, bitwise and logical operators, which are applied to the variables, and~~ wherein the graph comprises vertices corresponding to the operators, connecting the vertices corresponding to the variables.

107. (Previously presented) Apparatus according to claim 92, wherein the set of constraints is represented as a disjunction of multiple relations, with one of the hyper-arcs corresponding respectively to each of the relations, and wherein the processor is arranged to determine interim domains of the variables for each of the hyper-arcs, and to take a union of the interim domains for each of the variables to determine the output domains.

108. (Currently amended) Apparatus according to claim 92, wherein the processor is arranged to ~~determine the output domains~~ find the sets of values of the variables within the input domains such that for any given value in the respective output domain of each of the variables, there exist values of the other variables in the respective output domains thereof that, together with the given value, constitute a solution to the set of constraints.

109. (Currently amended) Apparatus according to claim 92, wherein the processor is arranged to ~~determine the output domains~~ find the sets of values of the variables within the input domains such that every set of values of the variables in the input domains that constitutes a solution to the set of constraints is contained in the output domains of the variables.

110. (Previously presented) Apparatus according to claim 92, wherein the set of constraints comprises a relation that relates to at least three of the variables.

111. (Canceled)

112. (Currently amended) Apparatus according to claim ~~111~~ 92, wherein the system comprises an electronic device, and wherein the inputs comprise commands and addresses to be input to the device.

113. (Currently amended) Apparatus according to claim 92, wherein the ~~set of variables comprises control parameters of~~ system comprises a mechanical system, and wherein the ~~processor is arranged to generate~~ inputs comprise a command to control the system ~~based on the output domains of the parameters~~.

114-117. (Canceled)

118. (Currently amended) A computer software product for ~~solving a constraint satisfaction problem~~ controlling operation of a target system, the product comprising a computer-readable medium in which program instructions are stored, which instructions, when read by a computer, cause the computer, upon receiving a set of variables that are characteristic of inputs to the target system, the variables having respective input domains, and a set of constraints comprising one or more relations defined as a combination of operators and expressed in a generic relational language, the combination comprising at least one operator selected from a group of arithmetic and bitwise operators, which are applied to the variables, to automatically parse the constraints expressed in the generic relational language so as to build a network ~~of~~ comprising one or more hyper-arcs

~~representing the set of constraints~~, each hyper-arc corresponding to ~~one of the relations expressed in terms of the operators and~~ a respective relation and comprising a constraint sub-network linking nodes in the network corresponding to the variables to which the operators are applied, to process each constraint sub-network so as to find sets of values of the variables within the input domains that satisfy the respective relation, ~~and~~ to reduce the input domains of the variables in the network ~~responsive to the operators~~ using the sets of values of the variables, so as to determine respective output domains of the variables that are consistent with the set of constraints, and to determine values of the inputs to be made to the target system based on the consistent output domains of the variables.